REMARKS

The Examiner continues to use any one of Satterfield, Kang, Park, Fujiwara or Koopman to reject claims 1 and 8 as having been anticipated.

Applicants respectfully disagree. None of the references cited by the Examiner discloses an instruction, as claimed in Applicants' claims.

Claims 1 and 8 have been amended to recite "in a single clock cycle, concatenating..., shifting..., and storing," or similar language. This feature is not disclosed or suggested in Satterfield, Kang, Park, Fujiwara or Koopman, and thus claims 1 and 8, as amended, were not anticipated by or would have been obvious from Satterfield, Kang, Park, Fujiwara or Koopman.

First, Satterfield discloses an encryption/decryption scheme. Kang discloses an apparatus for encoding variable length codes. Park discloses a decoder/encoder. Fujiwara discloses a codeword encoder. Koopman discloses a pseudorandom number generator and cryptographic authentication.

An instruction is an order given to a computer processor by a computer program. A processor is the logic circuitry that responds to and processes the basic instructions that drive a computer. None of these cited references disclose an instruction and thus, applicants' claims cannot be anticipated by (or would have been obvious from) Satterfield, Kang, Park, Fujiwara or Koopman.

Among the advantages, as stated in applicants' specification:

> The double shift instruction concatenates two long words (i.e., two 32 bit words) and shifts the result and saves the result as a longword. In the double shift instruction, the upper A-op shifts into lower B-op, with a "left rotate" of zero giving a zero shift (otherwise zero amount signifies indirect shift). The double shift instruction loads a destination register with a 32-bit longword that is formed by concatenating the A operands and B operands together, right shifting the 64-bit quantity by the specified amount, and storing the lower 32 bits.

Second, assuming arguendo, that the cited references did teach an instruction (and they do not), each of the cited references uses multiple clock cycles in multiple steps to accomplish multiple tasks.

For example, Satterfield discloses:

> Now looking at the bottom portion of FIG. 16, the result of two IDB operations (here M1 is the same width as M2) is stored in a 16 bit Shift Register 570. The two operations as shown in this figure are ADD 110, but in another implementation, not shown, could include other logic/mathematic operators. However the 16 bits of IDB are modified, they are stored in Shift Register 570, where similar to the discussion about the top portion of this figure, only a portion of the 16 bits are modified by the XOR 574. This XOR (574) has the effect of modifying the top four bits of the IDB byte (8 bits) in the lower half of 576 and the lower 4 bits in the IDB byte in the upper half of 576. The output 577 of the Shift Register 576 are moved 8 bits at a time to EDB 579 whose output 580 is placed into an output buffer in a normal fashion as per prior discussions.

(Col. 21, lines 65-67; col. 22, lines 1-12) These functions take Satterfield multiple cycles to perform and are not done with a single instruction in a single clock cycle.

Kang discloses:

> As shown in FIG. 3, if a first 8-bit segment "111X XXXX" from the second register 28 and a second 8-bit segment "1000 0011" from the fourth register 34 are inputted over parallel leads 25 and 35, the first barrel shifter 32 is responsive to a codeword length signal "M", i.e., M=3, on the lead 31 produced at the third register 30, to form an 8-bit window on its 16-bit inputs. The position of the 8-bit window is determined by shifting the window by M bits from the left-hand side of the 16-bit inputs. That is, the selection of M bits out of the first segment "111X XXXX" from the left-hand side and another selection of (8-M) bits out of the second segment "1000 0011" from the right-hand side remove the five meaningless bits of the first segment "111X XXXX", thereby to concatenate the variable-length codewords. After forming the window, the first barrel shifter 32 produces an 8-bit window output segment "0001 1111" to the fourth register 34 via the lead 33.

(Col. 5, lines 8-26) These functions take Kang multiple steps to perform and are not done with a single instruction in a single clock cycle.

In an identical fashion, Park, Fujiwara and Koopman disclose multiple clock cycles in multiple steps to accomplish multiple tasks, and not a single instruction done in a single clock cycle. Using multiple clock cycles in multiple steps to accomplish multiple tasks is different from "in a single clock cycle, concatenating..., shifting..., and storing." Accordingly, claims 1 and 8 were not anticipated by or would have been obvious from Satterfield, Kang, Park, Fujiwara or Koopman.

Dependent claims 2-7 and 9-14 are patentable for at least the same reasons as claims 1 and 8.

Applicants have added new claims 15-20. New claim 15 is an independent claim.

New claim 15 recites "in a single cycle in a parallel hardware-based multithreaded processor, concatenating..., right shifting..., and storing..."

For at least the same reasons as discussed with reference to claims 1 and 8, new claim 15 was not anticipated by or would have been obvious from Satterfield, Kang, Park, Fujiwara or Koopman.

New dependent claims 16-20 are patentable for at least the same reasons as claims 1, 8 and 15.

The fact that the applicants have not responded to any stated positions by the Examiner should not be construed as a concession by the applicants of those positions. The inclusion by the applicants of arguments for patentability should not be construed as a concession by the applicants that there are not other good reasons for patentability of those claims or other claims.

Applicant asks that all claims be examined in view of the amendment to the claims.

An RCE and an appropriate fee are enclosed herewith.

Please apply any other charges or credits to deposit account 06-1050.

Respectfully submitted,

Date: September 15, 2003

Kenneth F. Kozik
Reg. No. 36,572

Fish & Richardson P.C.
225 Franklin Street
Boston, MA 02110-2804
Telephone: (617) 542-5070
Facsimile: (617) 542-8906